FPGA IMPLEMENTATION OF DATA HIDING IN AUDIO SIGNAL USING LSB ENCODING

Ms.N.Gangarani, (gangarani21@gmail.com),Assistant Professor,AVS Engineering College,Salem, Tamilnadu.
Mr. G.Kanagaraj, (raj.techstorm@gmail.com), Assistant Professor,AVS Engineering College,Salem, Tamilnadu.
M.Dhanam,(eceam.dhanam@gmail.com)PG Scholar,AVS Engineering College,Salem, Tamilnadu.

ABSTRACT

Data hiding was done using plain text, still images, video and IP datagram for a lengthy era. In recent time’s audio steganography is spot of heart. An original method of top secret data to be unseen in acoustic by means of cryptography and steganography collectively. The protection of this method is enhanced by using of an encryption method prior to the data embedding step. First data is scientifically encrypted and fixed in audio. The perceptual excellence of the host audio signal was not to be degraded while embedding. The main goal of Text data hiding in Audio signal is to hide messages inside the audio in a way that does not allow any enemy to even detect that there is a second secret text message present in the audio. It can also be used for inserting hidden data into audio files for the authentication of spoken words and other sounds and for monitoring of the song over broadcast radio. Data hiding in audio signal architecture is going to design using Verilog HDL and simulation will be done by Modelsim software and synthesize will be done by Altera Quartus II software.

I. Introduction

Steganography is the art and science of hiding information by embedding messages within other, seemingly harmless messages. Steganography means “covered writing” in Greek. As the goal of steganography is to hide the presence of a message and to create a covert channel, it can be seen as the complement of cryptography, whose goal is to hide the content of a message. The goal of Steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second secret message present.” By using this proposed algorithm, I can hide our file of any format in an image and audio file. We can then send the image via e-mail attachment or post it on the web site and anyone with knowledge that it contains secret information, and who is in possession of the encryption password, will be able to open the file, extract the secret information and decrypt it. Steganography literally means covered writing. Its goal is to hide the fact that communication is taking place. In the field of Stenography, some terminology has been developed. The term cover is used to describe the original, innocent message, data, audio, still, video and so on. The network security is becoming more important as the number of data being exchanged on the Internet increases. Therefore, the confidentiality and data integrity are required to protect against unauthorized access. This has resulted in an explosive growth of the field of information hiding. In addition, the rapid growth of publishing and broadcasting technology also requires an alternative solution in hiding information. The copyright of digital media such as audio, video and other media available in digital form may lead to large-scale unauthorized copying. This is because the digital formats make it possible to provide high image quality even under multi-copying.

As technology scales into the nano scale regime, reliability is becoming a major challenge for on-chip interconnects. Interconnect reliability issues are caused by manufacturing defects or a variety of noise sources, such as external radiation crosstalk coupling supply voltage fluctuations. process

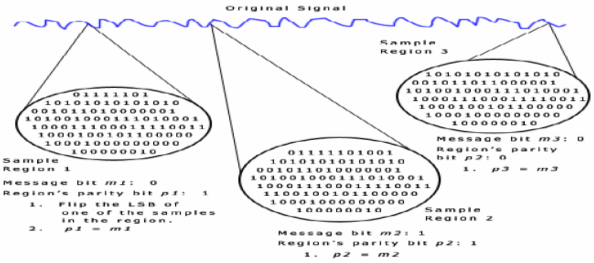


Fig. 1 parity coding procedure

II.Literature Review

Reversible data hiding (RDH) can extract secret messages and restore the original image without distortion. The reversibility benefits many practical applications such as medical image processing and multimedia archive management. Because of high image quality, histogram modification is applied to RDH in many literatures. In this paper, two-dimensional histogram and prediction-error expansion, are integrated with a well-designed difference-pair mapping (DPM) to improve embedding capacity. According to the simulation results, the proposed RDH scheme outperforms existing approaches, which are also based on two-dimensional histogram. On the average, image quality is improved by 3dB with the same embedding capacity. Maintaining at the same image quality, embedding capacity can be improved by approximately 30,000 bits.

III Existing System

Many mesh file formats (OFF, PLY, OBJ, VRML, X3D, etc.) are based on an indexed data structure. Triangle meshes are composed of two components: vertex data and connectivity (face) data. Vertex data include the positional coordinates of the vertices and, optionally, photometric information such as normal vectors or colors. The connectivity data supply the topological information that specifies which vertices belong to each triangle. Let f**v***igiN*=0 represent the sequence of vertices encountered as a mesh is being traversed, where **v***i* = (*vi*;*x*; *vi*;*y*; *vi*;*z*) and *N* is the number of vertices. Note that each coordinate *vi*; *j* < 1; *j* 2 f*x*; *y*; *z*g. Uncompressed representations of mesh models typically specify each vertex coordinate as a 32-bit floating-point number, and the number of each vertex coordinate's significant digit is 6In 1995, Deering advised that most applications do not require this level of precision and presented a lossy vertex data compression scheme. Positions are first normalized within an axisaligned bounding box. The coordinates are then uniformly quantized to *k* bits of precision so they can be represented as integers between 0 and 2*k* - 1. Empirically, *k* 2 [1; 33]. For each coordinate *vi*; *j* of a vertex, it is normalized to *v*0 *i*; *j*, *v*0 *i*; *j* = b*vi*; *j* × 10*k*c; *j* = *x*; *y*; *z*: Therefore, we implement encryption, decryption, data hiding, and message extraction based on *v*0 *i*; *j*. To generate processed meshes, we reversibly transform the processed integral coordinates of vertices ¯ *v*0 *i*; *j* to decimal coordinates ¯ *vi*; *j*, *v*¯ *i*; *j* = *v*¯ 0 *i*; *j*=10*k*; *j* = *x*; *y*; *z* Computer processors are often designed to process a data group into words of a given length of bits (8, 16, 32, 64 bits, etc.). When we manipulate integral Pre-processed coordinates, The value of *k* influences the time cost of each phase of the method, including encryption, data hiding, data extraction, decryption, and mesh recovery.

a) Encryption

We assume that we have pre-processed vertices, and denote the bits of a component of a vertex as *bi*; *j*;0; *bi*; *j*;1; :::; *bi*; *j*;*k*,where 1 _ *i* _ *N* and *j* 2 f*x*; *y*; *z*g. This implies that *bi*; *j*;*u* = b*v*0 *i*; *j*=2*u*c mod 2; *u* = 0; 1; *k*: The content owner then chooses an encryption key *K*1 to generate pseudo-random bits using a stream cipher function (e.g., RC4 orSEAL), and encrypts the bitstream of the preprocessed mesh where *ki*; *j*;*u* are the key stream bits, *ei*; *j*;*u* are the generated cipher text. Accordingly, the encrypted integral mesh model can be constructed by *Ei*; *j* = *k*X*u*=1 where *Ei*; *j* are the integral value of coordinates, 1 _ *i* _ *N* and *j* 2 f*x*; *y*; *z*g. Note that the stream only scrambles coordinate values, but does not shuffle coordinate locations**.**

b) Data Embedding

With the encrypted data, although a data hider does not know the original mesh content, he can embed an additional message into the mesh by modifying a small proportion of the encrypted data. The selection of mesh parts that are employed to embed messages is based on the following criterion. Since a vertex is contained within several triangles, once the vertex is modified to embed messages, the adjacent vertices should not be modified and are used to recover the central vertex by adjacent correlation at the receiver side.

c) Direct Decryption

For an encrypted mesh containing embedded data, a receiver first generates *ki*; *j*;*u* according to the encryption key *K*1, and calculates the exclusive OR of the received data and *ki*; *j*;*u* to decrypt the mesh model. We denote the decrypted bits as *b*0 *i*; *j*;*u*. Clearly,the original 32 - *m* most-significant bits (MSBs) are retrieved correctly. For a certain vertex, if the embedded bit in thelocal region including the vertex is zero and the vertex belongs to S*e*, the data hiding does not affect any encrypted bits of the vertex in the local region. Thus, the *u* decrypted LSBs must be the same as the original LSBs, implying that the decrypted integral value of the mesh coordinate is correct.

d) Data Extraction and Lossless recovery

If the receiver has the data-hiding key *K*2, the receiver will extract the embedded bits and recover the original mesh content from the direct decrypted mesh. We exploit spatial correlation between neighbouring coordinates to achieve a high embedding rate. Since the kind of 3D meshes constitutes a series of flat triangles adjacent to each other around the inspection point, angle deficit curvature is adopted to implement curvature measurement, which makes it unnecessary to use explicit derivative estimates. Meanwhile, the constraint of side length is taken as a part of the measurement of spatial correlation.

IV. Proposed System

The main goal of text data hiding in audio signal is to hide messages inside the audio in a way that does not allow any enemy to even detect that there is asecond secret text message present in the audio using lsb encoding technique. it can also be used for inserting hidden data into audio files for the authentication of spoken words and other sounds and for monitoring of the song over broadcast radio. text data is hidden in audio file without disturbing the quality of the audio file.

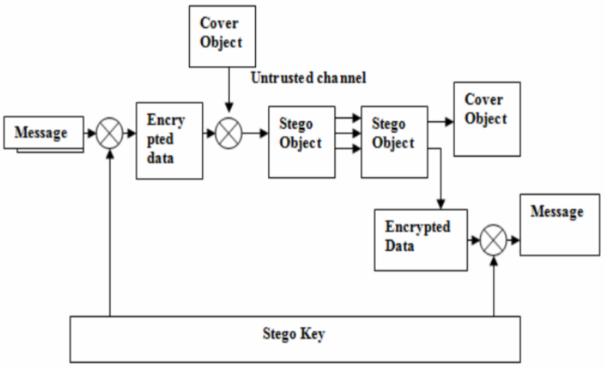
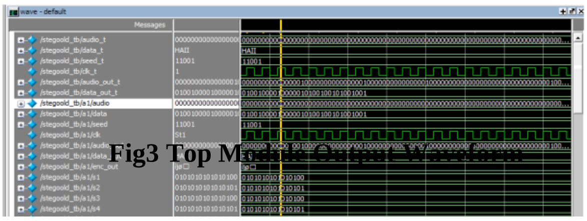


Figure 2. Block diagram of proposed system

an audio file format is a file format for storing audio data on a computer system. it can be a raw bit stream, but it is usually a container format or an audio data format with defined storage layer. the general approach towards storing digital audio is to sample the audio voltage which, on playback, would correspond to a certain level of signal in an individual channel with a certain resolution the number of bits per sample in regular intervals (forming the sample rate). this data can then be stored uncompressed, or compressed to reduce the file size. an lfsr is a shift register that, when clocked, advances the signal through the register from one bit to the next most-significant bit. some of the outputs are combined in exclusive-or configuration to form a feedback mechanism. a linear feedback shift register can be formed by performing exclusive-or on the outputs of two or more of the flip-flops together and

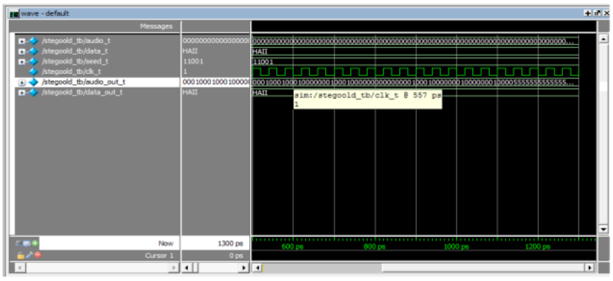
feeding those outputs back into the input of one of the flip-flops. Linear feedback shift registers make extremely good pseudorandom pattern generators. When the outputs of the flip-flops are loaded with a seed value (anything except all 0s, which would cause the LFSR to produce all 0 patterns) and when the LFSR is clocked, it will generate a pseudorandom pattern of 1s and 0s. Note that the only signal necessary to generate the test patterns is the clock. LSB Coding Least significant bit (LSB) coding is the simplest way to embed information in a digital audio file. By substituting the least significant bit of each sampling point with a binary message, LSB coding allows for a large amount of data to be encoded. In LSB coding, the ideal data transmission rate is 1 kbps per 1 khz. in some implementations of lsb coding, however, the least significant bit of a sample is replaced with a message bit. one should consider the signal content before deciding on the lsb operation to use. for example, a sound file that was recorded in a bustling subway station would mask low-bit encoding noise. on the other hand, the same noise would be audible in a sound file containing a piano solo. to extract a secret message from an lsb encoded sound file, the receiver needs access to the sequence of sample indices used in the embedding process. normally, the length of the secret message to be encoded is smaller than the total number of samples in a sound file.

V RESULTS



```
/stegoold_tb/audio_t        0000000000000000
/stegoold_tb/data_t         HAII
/stegoold_tb/seed_t         11001
/stegoold_tb/clk_t          1
/stegoold_tb/audio_out_t    00000000000000010
/stegoold_tb/data_out_t     010010000100000010
/stegoold_tb/a1/audio       0000000000000000
/stegoold_tb/a1/data        010010000100000010
/stegoold_tb/a1/seed        11001
/stegoold_tb/a1/clk         St1
/stegoold_tb/a1/audio...    0000000000000000
/stegoold_tb/a1/data_...    HAII
/stegoold_tb/a1/enc_out     ijø□
/stegoold_tb/a1/s1          0101010101010100
/stegoold_tb/a1/s2          0101010101010101
/stegoold_tb/a1/s3          0101010101010100
/stegoold_tb/a1/s4          0101010101010101
```

Fig3 Top Module Output Waveform

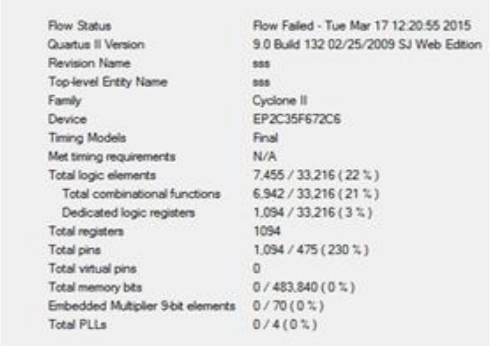


```
/stegoold_tb/audio_t        0000000000000000
/stegoold_tb/data_t         HAII
/stegoold_tb/seed_t         11001
/stegoold_tb/clk_t          1
/stegoold_tb/audio_out_t    000100010001000000
/stegoold_tb/data_out_t     HAII

sim:/stegoold_tb/clk_t @ 557 ps
1

Now        1300 ps
Cursor 1   0 ps
```

Fig4 Submodule Output Waveform



```
Flow Status                              Flow Failed - Tue Mar 17 12:20:55 2015
Quartus II Version                       9.0 Build 132 02/25/2009 SJ Web Edition
Revision Name                            sss
Top-level Entity Name                    sss
Family                                   Cyclone II
Device                                   EP2C35F672C6
Timing Models                            Final
Met timing requirements                  N/A
Total logic elements                     7,455 / 33,216 ( 22 % )
    Total combinational functions        6,942 / 33,216 ( 21 % )
    Dedicated logic registers            1,094 / 33,216 ( 3 % )
Total registers                          1094
Total pins                               1,094 / 475 ( 230 % )
Total virtual pins                       0
Total memory bits                        0 / 483,840 ( 0 % )
Embedded Multiplier 9-bit elements       0 / 70 ( 0 % )
Total PLLs                               0 / 4 ( 0 % )
```

Fig5 Area Report

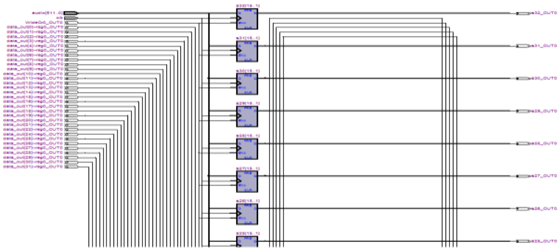


Fig.6Hardware Generation Report

a) COMPARISON TABLE

EXISTING SYSTEM	IMAGE	REVERSIBLE IMAGE DATA HIDING	9,250 LOGIC ELEMENTS 1708 – LOGIC REGISTERS
PROPOSED SYSTEM	AUDIO	LSB ENCODING	7837 – LOGIC ELEMENTS 1094 – LOGIC REGISTERS

VI. CONCLUSION

This system is to provide a good, efficient method for hiding the data from hackers and sent to the destination in a safe manner. This proposed system will not change the size of the file even after encoding and also suitable for any type of audio file format. Thus we conclude that audio data hiding techniques can be used for a number of purposes other than covert communication or deniable data storage, information tracing and finger printing, tamper detection. Thus I have successfully inserted and recovered the hidden data in an audio file. Audio file is manipulated in a way that may be detected by the receiver with a proper key. Thus Text data is hidden in audio file without disturbing the quality of the audio file. Another way to embed is to pad the secret message with random bits for that the length of the message is equal to the total number of samples. This increases the Probability that a would-be attacker will suspect secret communication.

REFERENCES

[1]. B. Ou, X. Li, Y. Zhao, R. Ni, and Y.-Q. Shi, "Pairwise prediction-error
expansion for e cient reversible data hiding," IEEE Transactions on Image
Processing, vol. 22, no. 12, pp. 5010–5021, 2013.
[2]. C. D. Vleeschouwer, J. F. Delaigle, and B. Macq, "Circular interpretation
of bijective transformations in lossless watermarking for media asset management," IEEE Transactions on Multimedia, vol. 5, pp. 97–105, March
2003.
[3]. Dalal N. Hmood, Khamael A. Khudhiar and Mohammad S. Altaei (2012).
A New Steganographic Method for Embedded Image In Audio File.
International Journal of Computer Science and Security(IJCSS) 6(2): pp.135-
141.
[4]. D.R.DenslinBrabin and Dr.J.JebamalarTamilselvi. "Reversible data
hiding: a survey," International Journal of Innovative Research in Computer
and Communication Engineering Vol. 1, Issue 3, May 2013
[5]. F. Willems, D. Maas, and T. Kalker, "Semantic lossless source coding,"
in Proc. 42nd Annu. Allerton Conf. Communication, Control and Computing,
2004.
[6]. H. S. Malvar and D. A. Florˆencio, "Improved spread spectrum: A new
modulation technique for robust watermarking," IEEE transactions on signal
processing, vol. 51, no. 4, pp. 898–905, 2003.
[7]. K. Wang, G. Lavou´e, F. Denis, and A. Baskurt, "Three-dimensional
meshes watermarking: Review and attack-centric investigation," in
International Workshop on Information Hiding, pp. 50–64, Springer, 2007.